

**IN THE CLAIMS:**

1 1. (CANCELLED).

1 2. (PREVIOUSLY PRESENTED): The method of claim 6 further comprising the step  
2 of defining a synchronization database having one or more synchronization records at the  
3 active supervisor, wherein  
4 the synchronization records store state information to be transmitted to the  
5 standby supervisor.

1 3. (ORIGINAL): The method of claim 2 further comprising the step of updating one or  
2 more of the synchronization records in response to an operating change at the at least one  
3 application program.

1 4. (ORIGINAL): The method of claim 3 wherein the transmitting step comprises the  
2 step of sending the one or more updated synchronization records to the standby supervi-  
3 sor.

1 5. (CANCELLED).

1 6. (PREVIOUSLY PRESENTED): In an intermediate network device having at least  
2 one line card defining a plurality of ports for receiving and forwarding messages and two  
3 or more supervisors, each supervisor configured to run one or more applications to facili-  
4 tate message handling by the network device, a method for continuing operation of at  
5 least one application despite a crash or failure, the method comprising the steps of:

6           designating a first supervisor to be an active supervisor and a second supervisor to  
7   be a standby supervisor for the network device;  
8           executing the at least one application at the active supervisor;  
9           holding the at least one application at the standby supervisor in a dormant state;  
10          transmitting state information generated during execution of the at least one ap-  
11   plication from the active supervisor to the standby supervisor;  
12          storing the state information at the standby supervisor;  
13          in response to the crash or failure at the active supervisor, carrying on execution  
14   of the at least one application at the standby supervisor based upon at least some of the  
15   stored state information;  
16          generating a sequence number for use in instructing the at least one line card to  
17   change operating condition;  
18          sending the sequence number to the at least one line card with the instruction;  
19          storing the sequence number at the at least one line card;  
20          sending the sequence number to the standby supervisor;  
21          storing the sequence number at the standby supervisor; and  
22          in response to the crash or failure of the active supervisor, comparing the se-  
23   quence number stored at the standby supervisor with the sequence number at the at least  
24   one line card.

1   7. (PREVIOUSLY PRESENTED): The method of claim 6 further comprising the step  
2   of continuing operation of the at least one line card, following the crash or failure of the  
3   active supervisor, if the sequence number stored at the at least one line card is one of (a)  
4   less than or equal to or (b) greater than the sequence number stored at the standby super-  
5   visor.

1 8. (PREVIOUSLY PRESENTED): The method of claim 7 further comprising the step  
2 of resetting the at least one line card, following the crash or failure of the active supervi-  
3 sor, if the sequence number stored at the at least one line card is one of (a) greater than or  
4 (b) less than or equal to the sequence number stored at the standby supervisor.

1 9. (PREVIOUSLY PRESENTED): The method of claim 6 further comprising the steps  
2 of:

3 determining the validity of the state information stored at the standby supervisor  
4 following the crash or failure of the active supervisor; and

5 blocking the at least one application from utilizing state information determined  
6 to be invalid in its execution.

1 10. (PREVIOUSLY PRESENTED): In an intermediate network device having at least  
2 one line card defining a plurality of ports for receiving and forwarding messages and two  
3 or more supervisors, each supervisor configured to run one or more applications to facili-  
4 tate message handling by the network device, a method for continuing operation of at  
5 least one application despite a crash or failure , the method comprising the steps of:

6 designating a first supervisor to be an active supervisor and a second supervisor to  
7 be a standby supervisor for the network device;

8 executing the at least one application at the active supervisor;

9 holding the at least one application at the standby supervisor in a dormant state;

10 transmitting state information generated during execution of the at least one ap-  
11 plication from the active supervisor to the standby supervisor;

12 storing the state information at the standby supervisor;

13           in response to the crash or failure at the active supervisor, carrying on execution  
14   of the at least one application at the standby supervisor based upon at least some of the  
15   stored state information;

16           creating, at the active supervisor, an instance of an event in response to a request  
17   from an application;

18           providing the event instance to the requesting application for processing;

19           providing the event instance to any listening applications that have registered for  
20   the event for processing;

21           passing the event instance to the standby supervisor;

22           receiving notifications from the requesting and listening applications that they  
23   have completed their processing of the event instance;

24           passing the notifications to the standby supervisor; and

25           in response to receiving notifications from the requesting and all listening applica-  
26   tions, closing the event instance at the active and standby supervisors.

1   11. (PREVIOUSLY PRESENTED): The method of claim 10 further comprising the step  
2   of:

3           in response to the crash or failure of the active supervisor, determining whether  
4   one or more event instances passed to the standby supervisor remain open;

5           for a given event instance that remains open, identifying the requesting and listen-  
6   ing applications that have not completed their processing of the given event instance;

7           for each requesting and listening application that has not completed its processing  
8   of the given event instance, calling a recovery function defined by the respective applica-  
9   tion to handle the open event instance.

1 12. (PREVIOUSLY PRESENTED): An intermediate network device for use in a com-  
2 puter network, the network device comprising:

3 a first supervisor card in communicating relationship with the one or more line  
4 cards;

5 a second supervisor card in communication relations with the first supervisor  
6 card;

7 an application loaded onto the first and second supervisor cards, the application  
8 configured to define and manipulate a plurality of state variables;

9 at least one line card defining a plurality of ports for forwarding messages across  
10 the computer network, the at least one line card in communicating relationship with the  
11 first and second supervisor cards and configured to receive and maintain port state infor-  
12 mation from the application; and

13 a high availability entity disposed on both the first and second supervisor cards,  
14 the high availability entities comprising:

15 an event mechanism for notifying a selected one of the first or second su-  
16 pervisor cards of changes to the application's state variables;

17 a database mechanism for storing the state variables at the first and second  
18 supervisor cards; and

19 a sequence mechanism for ensuring that the state variables stored at the  
20 first and second supervisor cards are consistent with the port state information  
21 maintained at the at least one line card, the sequence mechanism resetting the at  
22 least one line card in the event that the state variables and the port state informa-  
23 tion differ after a failure of one of the first or second supervisor cards.

1 13. (ORIGINAL): The network device of claim 12 wherein:

2           the first supervisor card is designated as an active supervisor card and the second  
3 supervisor card is designated as a standby supervisor card;

4           the application is allowed to run on the active supervisor card but not on the  
5 standby supervisor card; and

6           in response to a crash or failure of the active supervisor card, the application car-  
7 ries on its execution from the standby supervisor card utilizing at least some of the state  
8 variables stored at the database mechanism of the standby supervisor card.

1   14. (CANCELLED).

1   15. (PREVIOUSLY PRESENTED) A method for operating a network device, compris-  
2 ing:

3           operating an active supervisor, the active supervisor receiving state information  
4 from at least one line card;

5           generating a sequence number by the active supervisor in response to receipt of  
6 the state information;

7           returning the sequence number to the at least one line card;

8           storing the state information and sequence number to a standby supervisor;

9           in response to failure of the active supervisor, switching control to the standby su-  
10 pervisor;

11          comparing, by the standby supervisor, a stored sequence number with a reported  
12 sequence number, the reported sequence number reported by a line card; and

13          resetting the line card if the reported sequence number is different than the stored  
14 sequence number.

1 16. (PREVIOUSLY PRESENTED) A method for operating a network device, compris-  
2 ing:

3 operating an active supervisor, the active supervisor creating an instance of an  
4 event in response to a change in operating state from a requesting application;

5 providing the event instance to the requesting application and any listening appli-  
6 cations that have registered for the event for processing;

7 passing the event instance to a standby supervisor;

8 receiving notifications from the requesting and listening applications that they  
9 have completed their processing of the event instance;

10 passing the notifications to the standby supervisor; and

11 in response to receiving notifications from the requesting and all listening applica-  
12 tions, closing the event instance at the active and standby supervisors.

1 17. (PREVIOUSLY PRESENTED) The method of claim 16, further comprising:

2 in response to a failure of the active supervisor, determining whether one or more  
3 event instances passed to the standby supervisor remain open;

4 identifying the requesting and listening applications, if any, that have not com-  
5 pleted their processing of an open event instance; and

6 calling a recovery function defined by the respective application to handle the  
7 open event instance.

1 18. (PREVIOUSLY PRESENTED) A network device, comprising:

2 at least one line card;

3           an active supervisor, the active supervisor to receive state information from the at  
4   least one line card, generate a sequence number in response to receipt of the state infor-  
5   mation, and return the sequence number to the at least one line card;

6           a standby supervisor, the standby supervisor to store the state information and se-  
7   quence number, wherein if the active supervisor fails and control is switched to the  
8   standby supervisor, the standby supervisor is to compare a stored sequence number with a  
9   reported sequence number, the reported sequence number reported by a line card, and to  
10   reset the line card if the reported sequence number is different than the stored sequence  
11   number.

1   19. (PREVIOUSLY PRESENTED) A network device, comprising:

2           an active supervisor to run applications, the active supervisor to create an instance  
3   of an event in response to a change in operating state from a requesting application, to  
4   provide the event instance to the requesting application and any listening applications that  
5   have registered for the event for processing, and to receive notifications from the request-  
6   ing and listening applications that they have completed their processing of the event in-  
7   stance; and

8           a standby supervisor to receive the event instance and the notifications from the  
9   active supervisor, where in response to receiving notifications from the requesting and all  
10   listening applications, the active and standby supervisors are to close the event instance.

1   20. (PREVIOUSLY PRESENTED) The network device of claim 19, further compris-  
2   ing:

3           in response to a failure of the active supervisor, the standby supervisor is further  
4   to determine whether one or more event instances passed to the standby supervisor re-  
5   main open, to identify the requesting and listening applications, if any, that have not



6 completed their processing of an open event instance, and to call a recovery function de-  
7 fined by the respective application to handle the open event instance.

1 21. (PREVIOUSLY PRESENTED) A network device, comprising:

2 means for operating an active supervisor, the active supervisor receiving state in-  
3 formation from at least one line card;

4 means for generating a sequence number by the active supervisor in response to  
5 receipt of the state information;

6 means for returning the sequence number to the at least one line card;

7 means for storing the state information and sequence number to a standby super-  
8 visor;

9 means for switching control to the standby supervisor in response to a failure of  
10 the active supervisor;

11 means for comparing, by the standby supervisor, a stored sequence number with a  
12 reported sequence number, the reported sequence number reported by a line card; and

13 means for resetting the line card if the reported sequence number is different than  
14 the stored sequence number.

1 22. (PREVIOUSLY PRESENTED) A network device, comprising:

2 means for operating an active supervisor, the active supervisor creating an in-  
3 stance of an event in response to a change in operating state from a requesting applica-  
4 tion;

5 means for providing the event instance to the requesting application and any lis-  
6 tening applications that have registered for the event for processing;

7 means for passing the event instance to a standby supervisor;

8 means for receiving notifications from the requesting and listening applications  
9 that they have completed their processing of the event instance;

10 means for passing the notifications to the standby supervisor; and

11 means for in response to receiving notifications from the requesting and all listen-  
12 ing applications, closing the event instance at the active and standby supervisors.

1 23. (PREVIOUSLY PRESENTED) The network device of claim 22, further compris-  
2 ing:

3 in response to a failure of the active supervisor, means for determining whether  
4 one or more event instances passed to the standby supervisor remain open;

5 means for identifying the requesting and listening applications, if any, that have  
6 not completed their processing of an open event instance; and

7 means for calling a recovery function defined by the respective application to  
8 handle the open event instance.

1 24. (PREVIOUSLY PRESENTED) A computer readable media, comprising: the com-  
2 puter readable media containing instructions for execution on a processor for the practice  
3 of the method of,

4 operating an active supervisor, the active supervisor receiving state information  
5 from at least one line card;

6 generating a sequence number by the active supervisor in response to receipt of  
7 the state information;

8 returning the sequence number to the at least one line card;

9 storing the state information and sequence number to a standby supervisor;

10           in response to a failure of the active supervisor, switching control to the standby  
11 supervisor;

12           comparing, by the standby supervisor, a stored sequence number with a reported  
13 sequence number, the reported sequence number reported by a line card; and

14           resetting the line card if the reported sequence number is different than the stored  
15 sequence number.

1   25. (PREVIOUSLY PRESENTED) A computer readable media, comprising: the com-  
2 puter readable media containing instructions for execution on a processor for the practice  
3 of the method of,

4           operating an active supervisor, the active supervisor creating an instance of an  
5 event in response to a change in operating state from a requesting application;

6           providing the event instance to the requesting application and any listening appli-  
7 cations that have registered for the event for processing;

8           passing the event instance to a standby supervisor;

9           receiving notifications from the requesting and listening applications that they  
10 have completed their processing of the event instance;

11          passing the notifications to the standby supervisor; and

12          in response to receiving notifications from the requesting and all listening applica-  
13 tions, closing the event instance at the active and standby supervisors.

1   26. (PREVIOUSLY PRESENTED) The computer readable media of claim 25, further  
2 carrying instructions for the method of,

3           in response to a failure of the active supervisor, determining whether one or more  
4 event instances passed to the standby supervisor remain open;

5 identifying the requesting and listening applications, if any, that have not com-  
6 pleted their processing of an open event instance; and  
7 calling a recovery function defined by the respective application to handle the  
8 open event instance.

1 27. (PREVIOUSLY PRESENTED) Electromagnetic signals propagating on a computer  
2 network, comprising: the electromagnetic signals carrying instructions for execution on a  
3 processor for the practice of the method of,

4 operating an active supervisor, the active supervisor receiving state information  
5 from at least one line card;

6 generating a sequence number by the active supervisor in response to receipt of  
7 the state information;

8 returning the sequence number to the at least one line card;

9 storing the state information and sequence number to a standby supervisor;

10 in response to a failure of the active supervisor, switching control to the standby  
11 supervisor;

12 comparing, by the standby supervisor, a stored sequence number with a reported  
13 sequence number, the reported sequence number reported by a line card; and

14 resetting the line card if the reported sequence number is different than the stored  
15 sequence number.

1 28. (PREVIOUSLY PRESENTED) Electromagnetic signals propagating on a computer  
2 network, comprising: the electromagnetic signals carrying instructions for execution on a  
3 processor for the practice of the method of,

4           operating an active supervisor, the active supervisor creating an instance of an  
5 event in response to a change in operating state from a requesting application;

6           providing the event instance to the requesting application and any listening appli-  
7 cations that have registered for the event for processing;

8           passing the event instance to a standby supervisor;

9           receiving notifications from the requesting and listening applications that they  
10 have completed their processing of the event instance;

11          passing the notifications to the standby supervisor; and

12          in response to receiving notifications from the requesting and all listening applica-  
13 tions, closing the event instance at the active and standby supervisors.

1   29. (PREVIOUSLY PRESENTED) The electromagnetic signals of claim 28, further  
2 carrying instructions for the method of,

3           in response to a failure of the active supervisor, determining whether one or more  
4 event instances passed to the standby supervisor remain open;

5           identifying the requesting and listening applications, if any, that have not com-  
6 pleted their processing of an open event instance; and

7           calling a recovery function defined by the respective application to handle the  
8 open event instance.